

Weekly report (2013.6.3~6.9)

Done

- 1) I implement a pixel-based method to complete the preprocessing (by the way, reconstruct my code to be more clear). Then, I compare it to the old implement with 3 scenes(city, town, museum). To simplify the comparison, I fix the number of map tasks to 350 and reduce tasks to $35 \times 12 = 420$. The result is shown below:

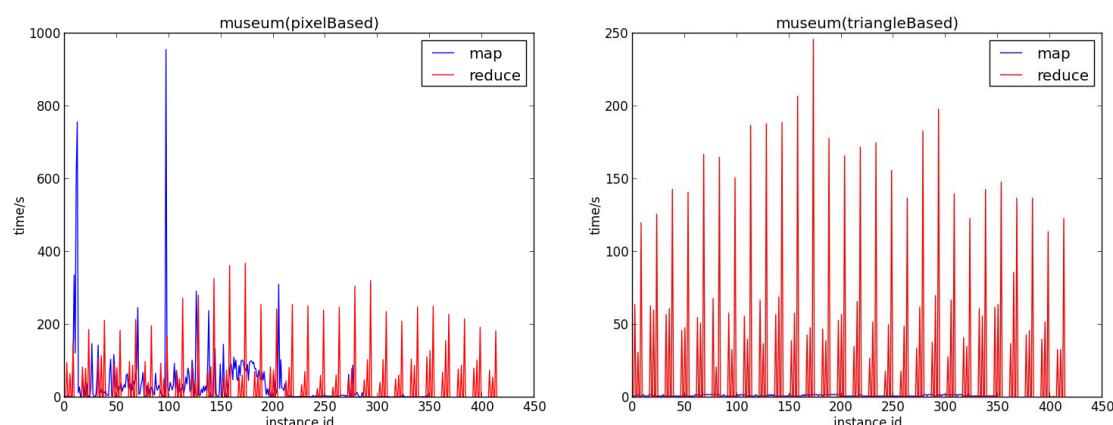
Figure 1 compare two method (pixel-based, triangle-based) with 3 scenes

scene	triangle number	resolution	triangle-based	pixel-based
	982126	640*480	51s	31s
		20480*15360	3m10s	10m13s
	1604502	640*480	2m30s	40s
		20480*15360	3m06s	10m52s
	12308935	640*480	41s	33s
		20480*15360	6m45s	28m27s

From the figure, we can see that at 640*480, pixel-based method is better, but at 20480*15360, triangle-based is better.

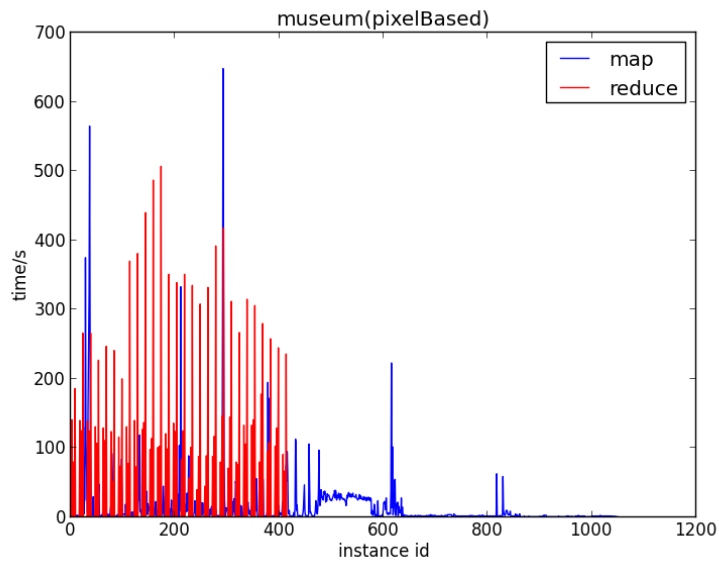
Then, I look into the execute time of each instance. For “museum”, it’s shown below:

Figure 2 instance time for museum in two method



We can find that there is apparent bottle-neck in the pixel-based method (some map task lasts a long time). So, I tried using more map tasks, as shown below, reduce time becomes longer while map tasks seem to get better balance. And it costs 23m25s.

Figure 3 museum preprocessed in pixel-based method with 350*3 map tasks



As a conclusion, this experiment shows that pixel-based method suits low resolution, while triangle-based method performs better at a higher resolution. As to the bottle neck occurred in the pixel-based case, I'll find some way to improve it.

To Do

- 1) Try to improve the bottle neck in pixel-based method and do experiment with boeing.